

APPLICATION
FOR
UNITED STATES LETTERS PATENT

TITLE: PRIORITY SERVICE PROCESS FOR SERVING DATA
UNITS ON A NETWORK

APPLICANT: GEPING CHEN

CERTIFICATE OF MAILING BY EXPRESS MAIL

Express Mail Label No. EL935340401US

I hereby certify under 37 CFR §1.10 that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Commissioner for Patents, Washington, D.C. 20231.

Date of Deposit

Signature

Typed or Printed Name of Person Signing Certificate

12-13-01
Leroy Jenkins

PRIORITY SERVICE PROCESS FOR SERVING DATA UNITS ON A NETWORK

CROSS-REFERENCE TO RELATED APPLICATION

5 This application claims priority to U.S. Provisional Patent Application No. 60/295,601, filed June 4, 2001, entitled "Non-Copying Buffer Handling For Porting A Protocol Stack To Drivers", the contents of which are hereby incorporated by reference into this application as if set forth herein in full.

TECHNICAL FIELD

10 This invention relates to serving data units on a network.

BACKGROUND

15 Asynchronous transfer mode (ATM) has been selected as the Committee on Civilian Industrial Technology (CCITT) standard for switching and multiplexing on Broadband-Integrated Service Digital Networks (B-ISDN). B-ISDN supports services with diversified traffic characteristics and Quality of Service (QoS) requirements, such as data transfer, telephone/videophone, high definition television (HDTV), multimedia conferencing, medical diagnosis and real-time control.

20 To efficiently utilize network resources through multiplexing, while providing the required QoS to supported multiple applications, the service to a specific application should be dependent on the QoS requirements of that application. QoS is typically described in terms of some measure of the delay or loss that data units of an application suffer over a transmission path from source to destination. Since the universal data unit of
25 an ATM network is the (fixed size) cell, QoS requirements for ATM are usually described in terms of some metric of the cell delay and cell loss.

 One objective of a priority service scheme is to deliver service that is as close as possible to the target QoS specified by the associated cell delay/loss metric. A priority service scheme can be defined in terms of a cell serving (i.e., transmitting) policy

specifying: (a) which of the arriving cells are admitted to network buffer(s) and/or (b) which of the admitted cells is served next from those buffer(s). The former type of priority service scheme is typically referred to as a "space-priority" scheme and impacts on the delivered cell loss metric. The latter type is typically referred to as a "time-priority" (or priority-scheduling) scheme and impacts on the delivered cell delay metric.

In the absence of priority service, the network load may be set at a potentially very low level in order to provide the most stringent QoS to all applications on the network. Considering that QoS requirements can differ substantially for different applications, a non-priority service scheme may result in severe underutilization of the network's resources. For instance, cell loss probability requirements can range from 10^2 to 10^{10} cells and end-to-end cell delay requirements for real-time applications can range from below 25 milliseconds (ms) to 1000 ms.

SUMMARY

In general, in one aspect, the invention is directed to a priority service scheme for serving data units on a network. This aspect includes queuing data units from a first application in a first buffer, queuing data units from a second application in a second buffer, moving data units from the second buffer to the first buffer following a predetermined delay, and serving data units from the first buffer. This aspect of the invention may include one or more of the following features.

The data units from the first application may have a higher priority for transmission on the network than the data units from the second application. The data units from the first application and the data units from the second application may be served from the first buffer on a first-come-first-served basis. Data units from the first application that exceed a first time delay may be discarded and data units from the second application that exceed a second time delay may be discarded. The second time delay may exceed the predetermined time delay.

The data units from the second buffer may be moved to an end of the first buffer after data units from the first application. A time to move the data units from the second buffer to the first buffer may be determined. A circular buffer, a pointer and a timer may be used to determine the time to move the data units from the second buffer to the first

buffer. Data units may be served from the second buffer when the first buffer is empty. The data units may include Asynchronous Transfer Mode (ATM) cells and the network may be an ATM network.

This summary has been provided so that the nature of the invention can be understood quickly. A detailed description of illustrative embodiments of the invention is set forth below.

DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of buffers for serving data units.

Fig. 2 is a block diagram of part of a controller for determining a timing at which the data units may be moved between the buffers of Fig. 1.

Fig. 3 is a graph showing axes used in a performance analysis of a method for serving the data units from the buffers.

Fig. 4 is a graph showing a renewal cycle for cells on the axes of Fig. 3.

Fig. 5 is a view of a computer on which the method for serving the data units from the buffers may be implemented.

DETAILED DESCRIPTION

One factor in ATM network design is providing diversified QoS to applications with distinct characteristics. This is of particular importance in real-time applications, such as streaming audio and video. Buffer management schemes can play a role in providing the necessary diversification through an ATM cell admission and service policy. A flexible priority service policy for two applications (classes) with strict - and in general distinct - deadlines and different deadline violation rates is described herein. The flexible service policy described herein utilizes a buffer management scheme to provide the requisite QoS to the different applications.

Consider two different classes of traffic (from, e.g., two different computing applications) sharing a transmission link between a transmitter (not shown) and a receiver (not shown). The link is slotted and capable of serving one data unit (in the case of ATM, a cell) per time slot. Let H (for high priority) and L (for low priority) denote the two

classes of traffic and let superscript X^Y denote a quantity associated with traffic class X^Y . A process 10 for serving ATM cells from a device, such as a router or general-purpose computer, to an ATM network, is as follows:

- (1) H-cells (i.e., cells of an application "H") join the "Head of the Line" (HoL) service class upon arrival. HoL cells are served according to the HoL priority policy; i.e., no service is provided to other cells unless no HoL cell is present. H-cells that experience a delay of more than T^H slots are discarded.
- (2) L-cells (i.e., cells of an application "L") are served according to D-HoL priority. That is, L-cells join the HoL service class (as fresh arrivals) only after they have waited for D ($D \geq 1$) time slots. Since the service policy is assumed to be work-conserving, L-cells may be served before they join the HoL class provided that no HoL class cells are present. L-cells that experience a delay of more than T^L slots are discarded. It is assumed that $T^L \geq D$.
- (3) Within each service class (HoL or waiting-to-join L-cells), cells are served according to the First-Come First Served (FCFS) policy. In the FCFS policy, the first cell to arrive at the buffer is the first cell to be output from the buffer.

The cell serving policy of process 10 may be implemented by considering time-stamps associated with each cell arrival. However, such an approach may not be realistic for a high-speed, low-management complexity switching system. A simpler implementation of the cell serving policy of process 10 is shown in Fig. 1.

Referring to Fig. 1, as H-cells arrive, the H-cells are immediately queued (i.e., stored) in H-buffer 12. If the occupancy of H-buffer 12 exceeds a predetermined threshold (in this embodiment, a threshold that corresponds to T^H), newly-arriving H-cells are discarded (e.g., cells for real-time video that arrive past their preset "playout" time may be discarded). Since H-buffer 12 is served under the HoL priority policy, the discarded H-cells are precisely the ones whose deadline T^H would be violated. Accordingly, losing those cells will have little impact on overall QoS.

L-cells are queued in L-buffer 14 as they arrive. In this embodiment, L-buffer 14 is served only when H-buffer 12 is empty. Otherwise, cells in L-buffer 14 that experience a delay D are moved to the end of the queue of H-buffer 12, provided that the occupancy of H-buffer 12 does not exceed $T^L - D$, in which case these L-cells are discarded. The L-cells are served from the H-buffer in turn. Again, since H-buffer 12 is served under the HoL priority policy, the discarded L-cells are the ones whose deadline T^L would be violated. Accordingly, losing those cells will have little impact on overall QoS.

The arrangement shown in Fig. 1 implements (deadline-violating) cell discarding through a simple buffer threshold policy, avoiding a more complex time-stamping approach. The capacity of H-buffer 12 and L-buffer 14 are equal to $\max(T^H, T^L - D)$ and $\min(T^L, DN_{\max})$, respectively; where N_{\max} is the maximum number of L-cell arrivals per slot. The threshold "d" in Fig. 1 is equal to $\min(T^H, T^L - D)$. If $\min(T^H, T^L - D) = T^H$, then any H-cells more than d cells in H-buffer 12 are discarded.

There are several ways of identifying the time to move L-cells from L-buffer 14 to H-buffer 12. For example, time-stamping may be used (i.e., examining cell time-stamps). Time-stamp-based sorting in every slot presents a level of complexity that may not be tolerable in a high-speed networking environment. For this reason, alternatives to time-stamp-based implementation approaches may be used. For example, a list of cell arrival times and a clock may be used to control cell movement.

In this embodiment, a circular buffer (registers) that records the number of cell arrivals per slot is used. Referring to Figs. 1 and 2, the L-buffer controller may be a microprocessor, microcontroller, or the like that includes (or uses) a pool of D registers 16, a timer T 18, and a pointer P 20. The timer counts from 0 to $D-1$, increasing its content by one (1) at the count of every slot. After $D-1$, the timer is reset at the next slot and then continues counting. The current content of the timer indicates the register where the number of L-cell arrivals during the current slot are registered.

The pool of registers 16 may be viewed as a circular structure and the timer T may be viewed as a rotating pointer pointing to a register to be used at a current time (Fig. 2) (although this is not necessarily its actual structure). The register visited by timer T contains the number of L-cells that have experienced a delay D in L-buffer 14. These

cells, which are at the head of L-buffer 14, are moved to H-buffer 12 and the number of new L-cell arrivals over the current slot are registered in this register.

The timer T identifies the time that L-cells are moved to H-buffer 12. A mechanism may also be used to determine changes in the content of the registers due to service provided to L-cells that are in L-buffer 14. The pointer P is used for this purpose. This pointer P points to the (non-zero-content) register containing the number of L-cells that are the oldest in L-buffer 14. When service is provided to L-buffer 14 — i.e., when the H-buffer 12 is empty — the content of the register pointed to by pointer P is decreased by one. The pointer P moves from a current register to a next non-zero-content register if the content of the current register becomes zero. This occurs if the content of the current register is one and service is provided to an L-cell or if the timer T visits this register, and thus the corresponding L-cells are moved to H-buffer 12. If there is currently no L-cell in L-buffer 14, pointer P equals timer T.

The following evaluates the diversified QoS provided to the two classes of traffic, L and H, by the cell serving policy of process 10. Here, QoS is defined in terms of the induced cell loss probability, noting that cell loss and cell deadline violation are identical events under process 10. The diversity in the QoS requirements for the two applications served under process 10 is represented by differences in cell delay deadlines and cell loss probabilities. By controlling the delay D before the L-cells qualify for service under the HoL priority, it is expected that the induced cell loss probability will be affected substantially. The effectiveness of the D parameter is demonstrated through numerical results. The approach can be modified to yield other QoS measures such as the average cell delay and the tail of the cell delay distribution.

Fig. 3 shows three axes, which are used in the performance analysis of process 10. The L-axis 22 is used to describe L-cell arrivals at the time they occur. The H-axis 24 is used to describe H-cell arrivals at the time they occur. The system axis 26 is used to mark the current time. In this example, cell arrival and service completions are assumed to occur at the slot boundaries.

The H-cell and L-cell arrival processes are assumed to be independent and governed by geometrically distributed (per slot) batches with parameters q^H and q^L , respectively. The probability that the batch sizes N^H and N^L are equal to n is given by

$$P^H(N^H = n) = (q^H)^n(1 - q^H), \quad P^L(N^L = n) = (q^L)^n(1 - q^L), \quad n = 0, 1, 2, \dots \quad (1)$$

and the mean H-cell (L-cell) arrival rate is given by

$$\lambda^H = \frac{q^H}{1 - q^H} \quad \left(\lambda^L = \frac{q^L}{1 - q^L} \right) \quad (2)$$

It is noted that the assumptions concerning the arrival processes considered above are not critical, since they can be changed in any number of ways. For instance, two-state Markov arrival processes may be considered and the resulting system may be analyzed by applying the same approach (but requiring increased computations). The subject cell serving policy is also applicable to arrival processes that are independent and identically distributed with arbitrarily distributed batch sizes. In this case, the maximum batch size affects the magnitude of the numerical complexity.

The following analysis is based on renewal theory. Let $n, n \in \mathbb{N}$, denote the current time, where \mathbb{N} denotes the set of natural numbers. At this time, n , the server is ready to begin the service of the next cell. Consider the following definitions

V_n: A random variable describing the current length (in slots) of the unexamined interval at the H-axis. That is, all H-cell arrivals before time $n - V_n$ have been served, since no H-cell arrival after time $n - V_n + 1$ has been considered for service by time n .

U_n: A random variable such that $U_n + V_n$ describes the current length of the unexamined intervals on the L-axis.

It is shown that $\{U_n, V_n\}_{n \in \mathbb{N}}$ is a Markov process. Let $\{s_k\}_{k \in \mathbb{N}}$ denote a sequence of time instants (slot boundaries) at which the system is empty; $\{s_k\}_{k \in \mathbb{N}}$ is a renewal sequence. Consider the following definitions.

- 5 Y_k : A random variable describing the length of the k th renewal cycle; Y will denote the generic random variable (associated with a renewal cycle).
- Y_k^H / Y_k^L : A random variable describing the number of H-cells/L-cells transmitted over the k th renewal cycle; Y^H / Y^L will denote the generic random variable.
- 10 L_k^H / L_k^L : A random variable describing the number of H-cells/L-cells lost over the k th renewal cycle; L^H / L^L will denote the generic random variable.

It can be shown that $Y_k = Y_k^H + Y_k^L + 1$, for $Y_k \geq 1$. (3)

15 Referring to Fig. 4, a renewal cycle is shown. Cells marked by "x" are the ones that are lost (due to violation of the associated deadline). Transmitted cells are shown on the system axis. In this example, $T^H=3$, $T^L=4$, and $D=2$. The renewal cycle begins at time t_0 , at which time no cell is present. The first slot of the renewal cycle is always idle (no transmission occurs). At time slot t_4 , one L-cell is served and the first L-cell is discarded due to the expiration of its deadline ($t_4 - t_1 = T^L$). In fact, at t^3 , the first two L-cells switch to H-buffer 12, but only one is served before its deadline. At t_5 , the L-cells that arrived at t_3 switch to H-buffer ($t_5 - t_3 = D$). The H-cell that arrived at t_5 is served before these L-cells. At t_9 , the L-cell is served since no H-cell is present. At t_{10} , no cell is present and the

20 renewal cycle ends. In this example, $Y_k=10$, $Y_k^H=6$, $Y_k^L=3$, $L_k^H=1$, and $L_k^L=3$.

25 The evolution of the process $\{U_n, V_n\}$ can be derived for the example shown in Fig. 4. Notice that when $U_n > D$, U_n points to the oldest unexamined slot that is to be served under the HoL priority. The associated L-cells have switched priority and are the oldest cells in the system. If $U_n < D$, V_n points to the oldest unexamined slot that is to be served under the HoL priority. If $U_n = D$, the oldest unexamined intervals in both axes

30

have the same priority and the selection is made according to a probabilistic. One possible evolution of $\{U_n, V_n\}$ corresponding to Fig. 4 is shown below:

- 5 $t_1: (0, 1)$
 $t_2: (0, 2) \rightarrow (1, 1)$
 $t_3: (1, 2)$
 $t_4: (1, 3) \rightarrow (2, 2)$
 $t_5: (1, 3) \rightarrow (2, 2) \rightarrow (1, 2) \rightarrow (2, 1)$
 $t_6: (2, 2)$
 $t_7: (1, 3) \rightarrow (2, 2) \rightarrow (1, 2)$
 $t_8: (1, 3)$
10 $t_9: (1, 3) \rightarrow (2, 2) \rightarrow (1, 2) \rightarrow (2, 1) \rightarrow (1, 1) \rightarrow (2, 0) \rightarrow (1, 0)$
 $t_{10}: (1, 1) \rightarrow (2, 0) \rightarrow (1, 0) \rightarrow (0, 0)$

4.2. Derivation of system equations

The following definitions will be used in the analysis:

- 15 $Y^H(i, j)$: A random variable describing the number of H-cells transmitted over the interval between a time slot at which $\{U_n, V_n\}$ is in state (i, j) and the end of the renewal cycle which contains this slot.
 $Y^L(i, j)$: A random variable defined as $Y^H(i, j)$ for L-cells.
 $I^H(I^L)$: An indicator function assuming the value 1 if an H-cell (L-cell) is present at the slot of H-axis (L-axis) under current examination (as pointed to by V_n or $U_n + V_n$); it assumes the value 0 otherwise.
 $\hat{I}^H(\hat{I}^L)$: An indicator function defined as $\hat{I}^H = 1 - I^H$ ($\hat{I}^L = 1 - I^L$).
20 $[i, j]^*$: It is a function which determines the next state of $\{U_n, V_n\}$ taking into consideration possible violation of T^L :

$$[i, j]^* = \begin{cases} (i, j), & i + j \leq T^L, \\ (T^L - j, j) & \text{otherwise.} \end{cases}$$

- m : It denotes the lowest possible value of random variable V_n ; $m = 0$ if $D \neq 0$ and $m = -1$ if $D = 0$.
 $I_{\{H\}}(I_{\{L\}})$: An indicator function associated with decisions regarding the slot to be examined next
25 when $U_n = D$. $I_{\{H\}} = 1(I_{\{L\}} = 1)$ if the oldest unexamined slot in H-axis (L-axis) is considered next, and $I_{\{H\}} = 0(I_{\{L\}} = 0)$ otherwise. This is a design parameter which can

impact on the induced cell losses. In Appendix A, the expected values of these functions ($\mu^H = P\{I_{\{H\}} = 1\}$, $\mu^L = P\{I_{\{L\}} = 1\}$) are derived under the assumption that all cells (from both classes) arrived over the slots to be examined when $U_n = D$ are equally likely to be selected. \bar{X} : Denotes $E\{X\}$, where $E\{\cdot\}$ is the expectation operator.

In the sequel, recursive equations are derived for the calculation of \bar{Y}^H and \bar{Y}^L . Then, similar equations are derived for the calculation of \bar{L}^H and \bar{L}^L . As it will be shown later, these quantities will yield the cell loss probabilities. Finally, the similar approach for the calculation of the average cell delay and cell delay tail probabilities is outlined at the end of this section.

It is easy to observe that no cell is transmitted in the first slot and thus, process $\{U_n, V_n\}$ actually starts from state $(0, 1)$. Thus,

$$Y^H = Y^H(0, 1), \quad Y^L = Y^L(0, 1). \quad (4)$$

A careful consideration of the evolution of the recursions presented below shows that when process $\{U_n, V_n\}$ reaches the state $(0, 0)$, the renewal cycle ends. For this reason,

$$Y^H(0, 0) = 0, \quad Y^L(0, 0) = 0 \quad (5)$$

to terminate the current cycle. Notice also that U_n can exceed $D + 1$ only if $V_n = T^H$.

Case A. $T^L \geq T^H$.

Case A.1. $T^H > j > 0$, $\min(T^L - j, D + 1) \geq i \geq m$.

1. $i < D$:

$$\begin{aligned} Y^H(i, j) &= I^H + Y^H([i + \hat{I}^H, j - 2\hat{I}^H + 1]^*), \\ Y^L(i, j) &= Y^L([i + \hat{I}^H, j - 2\hat{I}^H + 1]^*). \end{aligned} \quad (6)$$

2. $i = D + 1$:

$$\begin{aligned} Y^H(D + 1, j) &= Y^H([i - \hat{I}^L, j - \hat{I}^L + 1]^*), \\ Y^L(D + 1, j) &= I^L + Y^L([i - \hat{I}^L, j - \hat{I}^L + 1]^*). \end{aligned} \quad (7)$$

3. $i = D$:

$$\begin{aligned} Y^H(D, j) &= \{I^H + Y^H([i + \hat{I}^H, j - 2\hat{I}^H + 1]^*)\}I_{\{H\}} + Y^H([i - \hat{I}^L, j - \hat{I}^L + 1]^*)I_{\{L\}}, \\ Y^L(D, j) &= Y^L([i + \hat{I}^H, j - 2\hat{I}^H + 1]^*)I_{\{H\}} + \{I^L + Y^L([i - \hat{I}^L, j - \hat{I}^L + 1]^*)\}I_{\{L\}}. \end{aligned} \quad (8)$$

Case A.2. $j = T^H$, $T^L - T^H \geq i \geq m$.

1. $i < D$:

$$Y^H(i, T^H) = I^H + Y^H([i + 1, T^H - \hat{I}^H]^*), \quad Y^L(i, T^H) = Y^L([i + 1, T^H - \hat{I}^H]^*). \quad (9)$$

2. $i \geq D + 1$:

$$Y^H(i, T^H) = Y^H([i - 2\hat{I}^L + 1, T^H]^*), \quad Y^L(i, T^H) = I^L + Y^L([i - 2\hat{I}^L + 1, T^H]^*). \quad (10)$$

3. $i = D$:

$$\begin{aligned} Y^H(D, T^H) &= \{I^H + Y^H([i + 1, T^H - \hat{I}^H]^*)\}I_{\{H\}} + Y^H([i - 2\hat{I}^L + 1, T^H]^*)I_{\{L\}}, \\ Y^L(D, T^H) &= Y^L([i + 1, T^H - \hat{I}^H]^*)I_{\{H\}} + \{I^L + Y^L([i - 2\hat{I}^L + 1, T^H]^*)\}I_{\{L\}}. \end{aligned} \quad (11)$$

Case A.3. $j = 0, \min(T^L, D + 1) \geq i \geq 1$.

$$Y^H(i, 0) = Y^H([i - \hat{T}^L, 1 - \hat{T}^L]^*), \quad Y^L(i, 0) = I^L + Y^L([i - \hat{T}^L, 1 - \hat{T}^L]^*). \quad (12)$$

Case B. $T^L < T^H$.

The equations under this case are derived similarly and are presented in Appendix B. By applying the expectation operator to the above equations, the following systems of linear equations are obtained, details are presented in Appendix C:

$$\begin{aligned} \bar{Y}^H(i, j) &= a^H(i, j) + \sum_{(i', j') \in R_0} b^H(i, j, i', j') \bar{Y}^H(i', j'), \\ \bar{Y}^L(i, j) &= a^L(i, j) + \sum_{(i', j') \in R_0} b^L(i, j, i', j') \bar{Y}^H(i', j'), \end{aligned} \quad (13)$$

where $R_0 = \{(i, j) : m \leq i \leq T^L, 0 \leq j \leq T^H\}$. It should be noted that these systems of linear equations are extremely sparse: only two to four coefficients are not zero per equation. Thus, it can be solved efficiently by using an iterative approach. The computation complexity is of the order of DT^H . For T^H and $D < 100$, it takes less than a couple of hours to solve these equations in a SUN SPARC20 workstation. From the solution of these equations, \bar{Y}^H and \bar{Y}^L are obtained from (see (4))

$$\bar{Y}^H = \bar{Y}^H(0, 1), \quad \bar{Y}^L = \bar{Y}^L(0, 1). \quad (14)$$

The expected value of the number of H-cells (L-cells) lost over a renewal cycle, $\bar{L}^H(\bar{L}^L)$, is derived by following a similar approach. The following quantities need to be defined first.

$L^H(i, j)$ ($L^L(i, j)$): A random variable describing the number of H-cells (L-cells) discarded over the interval between a time slot at which $\{U_n, V_n\}$ is in state (i, j) and the end of the renewal cycle which contains this slot.

$S_{\{i, j\}}$: An indicator function assuming the value 1 if there is a possibility to discard an L-cell as a result of the service to be provided in the current slot (due to resulting violation of its deadline T^L):

$$S_{\{i, j\}} = \begin{cases} 1 & \text{if } i + j = T^L, \\ 0 & \text{otherwise.} \end{cases}$$

The equation for the derivation of L_n^H and L_n^L are similar to those for the derivation of Y_n^H and Y_n^L and are given below. Notice again that

$$L^H = L^H(0, 1), \quad L^L = L^L(0, 1) \quad (15)$$

and

$$L^H(0, 0) = 0, \quad L^L(0, 0) = 0. \quad (16)$$

Two cases need to be considered: $T^L \geq T^H$ and $T^L < T^H$.

Case A. $T^L \geq T^H$.

Case A.1. $T^H > j > 0, \min(T^L - j, D + 1) \geq i \geq m$.

1. $i < D$:

$$\begin{aligned} L^H(i, j) &= L^H([i + \hat{T}^H, j - 2\hat{T}^H + 1]^*), \\ L^L(i, j) &= I^H S_{\{i, j\}} N^L + L^L([i + \hat{T}^H, j - 2\hat{T}^H + 1]^*). \end{aligned} \quad (17)$$

2. $i = D + 1$:

$$\begin{aligned} L^H(D + 1, j) &= L^H([i - \hat{I}^L, j - \hat{I}^L + 1]^*), \\ L^L(D + 1, j) &= I^L S_{\{D+1, j\}} N^L + L^L([i - \hat{I}^L, j - \hat{I}^L + 1]^*). \end{aligned} \quad (18)$$

3. $i = D$:

$$\begin{aligned} L^H(D, j) &= L^H([i + \hat{I}^H, j - 2\hat{I}^H + 1]^*) I_{\{H\}} + L^H([i - \hat{I}^L, j - \hat{I}^L + 1]^*) I_{\{L\}}, \\ L^L(D, j) &= \{I^H S_{\{D, j\}} N^L + L^L([i + \hat{I}^H, j - 2\hat{I}^H + 1]^*)\} I_{\{H\}} \\ &\quad + \{I^L S_{\{D, j\}} N^L + L^L([i - \hat{I}^L, j - \hat{I}^L + 1]^*)\} I_{\{L\}}. \end{aligned} \quad (19)$$

Case A.2. $j = T^H, T^L - T^H \geq i \geq m$.

1. $i < D$:

$$\begin{aligned} L^H(i, T^H) &= I^H N^H + L^H([i + 1, T^H - \hat{I}^H]^*), \\ L^L(i, T^H) &= I^H S_{\{i, T^H\}} N^L + L^L([i + 1, T^H - \hat{I}^H]^*). \end{aligned} \quad (20)$$

2. $i \geq D + 1$:

$$\begin{aligned} L^H(i, T^H) &= I^L N^H + L^H([i - 2\hat{I}^L + 1, T^H]^*), \\ L^L(i, T^H) &= I^L S_{\{i, T^H\}} N^L + L^L([i - 2\hat{I}^L + 1, T^H]^*). \end{aligned} \quad (21)$$

3. $i = D$:

$$\begin{aligned} L^H(D, T^H) &= \{I^H N^H + L^H([i + 1, T^H - \hat{I}^H]^*)\} I_{\{H\}} \\ &\quad + \{I^L N^H + L^H([i - 2\hat{I}^L + 1, T^H]^*)\} I_{\{L\}}, \\ L^L(D, T^H) &= \{I^H S_{\{D, T^H\}} N^L + L^L([i + 1, T^H - \hat{I}^H]^*)\} I_{\{H\}} \\ &\quad + \{I^L S_{\{D, T^H\}} N^L + L^L([i - 2\hat{I}^L + 1, T^H]^*)\} I_{\{L\}}. \end{aligned} \quad (22)$$

Case A.3. $j = 0, \min(T^L, D + 1) \geq i \geq 1$.

$$\begin{aligned} L^H(i, 0) &= L^H([i - \hat{I}^L, 1 - \hat{I}^L]^*), \\ L^L(i, 0) &= I^L S_{\{i, 0\}} N^L + L^L([i - \hat{I}^L, 1 - \hat{I}^L]^*). \end{aligned} \quad (23)$$

Case B. $T^L < T^H$.

The equations under this case are derived similarly (see also Case B in the derivation of Y^H and Y^L) and are not presented due to space consideration.

By taking expectation operation to both sides of the equations, the following systems of linear equations are obtained:

$$\begin{aligned} \bar{L}^H(i, j) &= a'^H(i, j) + \sum_{(i', j') \in R_0} b^H(i, j, i', j') \bar{L}^H(i', j'), \\ \bar{L}^L(i, j) &= a'^L(i, j) + \sum_{(i', j') \in R_0} b^L(i, j, i', j') \bar{L}^L(i', j'), \end{aligned} \quad (24)$$

where R_0 and coefficients $b^H(i, j, i', j')$ and $b^L(i, j, i', j')$ are identical to those in system (13), and constants $a'^H(i, j)$ and $a'^L(i, j)$ are derived as the corresponding ones in the system in (13). Finally,

$$\bar{L}^H = \bar{L}^H(0, 1), \quad \bar{L}^L = \bar{L}^L(0, 1). \quad (25)$$

The cell loss probabilities P_{loss}^H for H-cells and P_{loss}^L for L-cells are obtained from the following expressions:

$$P_{\text{loss}}^H = \frac{\bar{L}^H}{\bar{Y}^H + \bar{L}^H}, \quad P_{\text{loss}}^L = \frac{\bar{L}^L}{\bar{Y}^L + \bar{L}^L}. \quad (26)$$

Notice that $\bar{Y}^H + \bar{L}^H$ is the average number of H-cells over a renewal cycle which is also given by $\lambda^H \bar{Y}$. Similarly, $\bar{Y}^L + \bar{L}^L$ is the average number of L-cells over a renewal cycle which is also given by $\lambda^L \bar{Y}$.

By invoking renewal theory, the rate of service provided to H-cell and L-cell streams — denoted by λ_s^H and λ_s^L , respectively — is given by

$$\lambda_s^H = \frac{\bar{Y}^H}{\bar{Y}^H + \bar{Y}^L + 1}, \quad \lambda_s^L = \frac{\bar{Y}^L}{\bar{Y}^H + \bar{Y}^L + 1}. \quad (27)$$

Alternatively, the cell loss probabilities — given by (26) — can be obtained as

$$P_{\text{loss}}^H = \frac{\lambda^H - \lambda_s^H}{\lambda^H}, \quad P_{\text{loss}}^L = \frac{\lambda^L - \lambda_s^L}{\lambda^L}. \quad (28)$$

Notice that computation of P_{loss}^H and P_{loss}^L from (27) and (28) does not require computation of L^H and L^L . It should be noted, however, that Eq. (28) can potentially introduce significant numerical error, especially if the cell loss rates are very low. For this reason, results have been obtained by invoking Eq. (26) in this paper.

As stated earlier, other measures of the QoS can be derived by following the above approach. The calculation of the average delay of the successfully transmitted cells and the tail of the delay probability distribution are outlined below. Equations similar to those presented for $L^H(i, j)$ and $L^L(i, j)$ can be derived, where the associated quantities of interest — instead of discarded H-cells in $L^H(i, j)$ and L-cells in $L^L(i, j)$ — are properly defined below.

- $C^H(i, j)$ ($C^L(i, j)$): A random variable describing the cumulative delay of successfully transmitted H-cells (L-cells) over the interval between a time slot at which $\{U_n, V_n\}$ is in state (i, j) and the end of the renewal cycle which contains this slot.
- $B_h^H(i, j)$ ($B_l^L(i, j)$): A random variable describing the number of H-cells (L-cells) which have experienced a delay less than or equal to h (l) over the interval between a time slot at which $\{U_n, V_n\}$ is in state (i, j) and the end of the renewal cycle which contains this slot.

Then,

$$C^H = C^H(0, 1), \quad C^L = C^L(0, 1), \quad B_h^H = B_h^H(0, 1), \quad B_l^L = B_l^L(0, 1). \quad (29)$$

The average delays for H-cells and L-cells are given by

$$\bar{D}^H = \frac{\bar{C}^H}{\bar{Y}^H}, \quad \bar{D}^L = \frac{\bar{C}^L}{\bar{Y}^L}. \quad (30)$$

The tail of the delay probability distribution is given by

$$P^H(D^H > h) = 1 - \frac{\bar{B}_h^H}{\bar{Y}^H + \bar{L}^H} = 1 - \frac{\bar{B}_h^H}{\lambda^H \bar{Y}}, \quad P^L(D^L > l) = 1 - \frac{\bar{B}_l^L}{\bar{Y}^L + \bar{L}^L} = 1 - \frac{\bar{B}_l^L}{\lambda^L \bar{Y}}. \quad (31)$$

To derive the quantities in (31), similar equations to those associated with $L^H(i, j)$ or $L^L(i, j)$ can be derived by replacing the first of the two right-hand side terms in those equations — counting discarded cells — by functions that count the total delay of the currently transmitted cell (in determining $C^H(i, j)$ or $C^L(i, j)$) or count the number of cells transmitted over the current slot which experienced a delay of less than or equal to h or l slots (in determining $B_h^H(i, j)$ or $B_l^L(i, j)$). These functions — denoted by $F_{\{i,j\}}^H$ ($F_{\{i,j\}}^L$) and $G_h^H(i, j)$ ($G_l^L(i, j)$), respectively — are given by the following:

$$F_{\{i,j\}}^H = \begin{cases} j & \text{if an H-cell is served,} \\ 0 & \text{otherwise,} \end{cases}$$

$$F_{\{i,j\}}^L = \begin{cases} i + j & \text{if an L-cell is served,} \\ 0 & \text{otherwise,} \end{cases}$$

$$G_h^H(i, j) = \begin{cases} 1 & \text{if an H-cell is served and } j \leq h, \\ 0 & \text{otherwise,} \end{cases}$$

$$G_l^L(i, j) = \begin{cases} 1 & \text{if an L-cell is served with } i + j \leq l, \\ 0 & \text{otherwise.} \end{cases}$$

Fig. 5 shows a server (computer) 30 on which process 10 may be executed. Server 30 includes a processor 32, a memory 34, and a storage medium 36 (e.g., a hard disk) (see view 38). Storage medium 36 stores machine-executable instructions 40, which are executed by processor 32 out of memory 34 to perform process 10 on incoming data units, such as ATM cells, to serve them to a network.

Process 10, however, is not limited to use with the hardware and software of Fig. 5; it may find applicability in any computing or processing environment. Process 10 may be implemented in hardware, software, or a combination of the two. Process 10 may be implemented in one or more computer programs executing on programmable computers or other machines that each includes a processor, a storage medium readable by the processor (including volatile and non-volatile memory and/or storage elements).